Amendments to the Specification:

Please amend the paragraph beginning at line 10 on page 7 with the following:

--The multiplexing arrangement 275 receives the information from the prioritized content-addressable-memory (CAM) arrangement 270, the physical register destination (PDst) from the allocator arrangement 265, and the source information of the instructions from the retirement overrides arrangement 260, and provides information (associated with, for example, an instruction) for assigning physical sources. Likewise, multiplexing arrangement [[280]] 285 receives the information from the prioritized content-addressable-memory (CAM) arrangement [[289]] 280, the physical register destination (PDst) from the allocator arrangement 265, and the source information of the instructions from the retirement overrides arrangement 260, and provides information (associated with, for example, another instruction) for assigning physical sources.--

Please amend the paragraph beginning at line 5 on page 13 with the following:

--In connection with the foregoing, the fast renaming apparatus, arrangement and/or method may be described as follows. If the starting address is fixed (as may be the case, for example, with a trace cache), the dependency chain information is believed to be relatively static across a cache line, but does depend on conditional branches. In particular, for a trace cache, the dependency information is almost all or essentially all static. A branch mis-predict may occur when the system mis-predicts the next instruction information, which ends the tracing of the instructions in a trace cache. Accordingly, the dependency information may be cached to eliminate the use of the prioritized content-addressable-memories (CAMs) with the renamer arrangement and/or method of Figure 2. Caching the dependency chain information is believed to be effective for providing sequential allocation, and may be done using the dependency information field arrangement 400 of Figure 4. In the field arrangement 400, the first source (Src1) 410 and the second source (Src2) 420 information are provided using three (3) bits and the destination field (Dst) 430 is provided using one (1) bit, which collectively provides the cached dependency chain information for use in the fast renaming apparatus, method and system of Figures 5 and 6.--

Amendments to the Claims:

Without prejudice, this listing of the claims replaces all prior versions and listings of the claims in the present application:

Listing of Claims:

1. (Original) A method for renaming a source for use with a processor, the method comprising:

providing at least one instruction;

building instruction dependency information based on the at least one instruction;

caching the at least one instruction with the instruction dependency information to provide cached instruction information;

renaming a register based on the cached instruction information to provide a renamed register; and

multiplexing the instruction dependency information and the renamed register to rename the source.

2. (Original) The method of claim 1, wherein the cached instruction information includes first source information, second source information and destination information.

3. (Original) The method of claim 2, wherein the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit.

4. (Original) The method of claim 1, wherein the first source information and the second source information denote a rename window instruction from which the at least one instruction depends.

5. (Original) The method of claim 1, wherein a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

6. (Original) The method of claim 1, wherein:

the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit;

the first source information and the second source information denote a rename window instruction from which the at least one instruction depends; and

a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

7. (Original) The method of claim 6, wherein the processor is a microprocessor.

8. (Original) A system for renaming a source for use with a processor, the system comprising:

a fetch and decoding arrangement for fetching and decoding at least one instruction from the processor;

a build-instruction-dependency arrangement for building instruction dependency information based on the at least one instruction;

an instruction cache arrangement for caching the at least one instruction with the instruction dependency information to provide cached instruction information, the build-instruction-dependency arrangement providing the instruction dependency information to the instruction cache arrangement;

a renamer arrangement for renaming a register based on the cached instruction information and for providing a renamed register; and

a multiplexing arrangement for multiplexing the instruction dependency information and the renamed register and for providing a renamed source.

9. (Original) The system of claim 8, wherein the cached instruction information includes first source information, second source information and destination information.

10. (Original) The system of claim 9, wherein the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit.

11. (Original) The system of claim 8, wherein the first source information and the second source information denote a rename window instruction from which the at least one instruction depends.

12. (Original) The system of claim 8, wherein a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

13. (Original) The system of claim 8, wherein:

the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit;

the first source information and the second source information denote a rename window instruction from which the at least one instruction depends; and

a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

14. (Original) The system of claim 13, wherein the processor is a microprocessor.

15. (Original) A system for renaming a source for use with a processor, the system comprising:

means for fetching and decoding at least one instruction from the processor;

means for building instruction dependency information based on the at least one instruction;

means for caching the at least one instruction with the instruction dependency information and for providing cached instruction information, the means for building instruction dependency information providing the instruction dependency information to the means for caching the instruction;

means for renaming a register based on the cached instruction information and for providing a renamed register; and

means for multiplexing the instruction dependency information and the renamed register and for providing a renamed source.

16. (Original) The system of claim 15, wherein the cached instruction information includes first source information, second source information and destination information.

17. (Original) The system of claim 16, wherein the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit.

18. (Original) The system of claim 15, wherein the first source information and the second source information denote a rename window instruction from which the at least one instruction depends.

19. (Original) The system of claim 15, wherein a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

20. (Original) The system of claim 15, wherein:

the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit;

the first source information and the second source information denote a rename window instruction from which the at least one instruction depends; and

a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

21. (Original) The system of claim 15, wherein the processor is a microprocessor.

22. (Original) A set of instructions residing in a storage medium, said set of instructions capable of being executed by a processor to implement a method for renaming a source for use with a processor, the method comprising:

providing at least one instruction;

building instruction dependency information based on the at least one instruction;

caching the at least one instruction with the instruction dependency information to provide cached instruction information;

renaming a register based on the cached instruction information to provide a renamed register; and

multiplexing the instruction dependency information and the renamed register to rename the source.

23. (Original) The set of instructions residing in a storage medium of claim 22, wherein the cached instruction information includes first source information, second source information and destination information.

24. (Original) The set of instructions residing in a storage medium of claim 23, wherein the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit.

25. (Original) The set of instructions residing in a storage medium of claim 22, wherein the first source information and the second source information denote a rename window instruction from which the at least one instruction depends.

26. (Original) The set of instructions residing in a storage medium of claim 22, wherein a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

27. (Original) The set of instructions residing in a storage medium of claim 22, wherein:

the first source information includes three bits, the second source information includes three bits, and the destination information includes one bit;

the first source information and the second source information denote a rename window instruction from which the at least one instruction depends; and

a virtual ID is formed by at least one of adding, combining, concatenating or logically OR-ing upper bits from a renamer with lower bits from an instruction cache.

28. (Original) The set of instructions residing in a storage medium of claim 27, wherein the processor is a microprocessor.

29. (New) The method of claim 1, wherein the instruction dependency information indicates upon which other instruction the at least one instruction depends.

30. (New) The method of claim 1, wherein the renamed register is a virtual renamed register.

31. (New) The method of claim 30, wherein the virtual renamed register refers to a value that is to be produced in the future by the at least one instruction.

32. (New) The method of claim 2, wherein the first and second source information includes exactly three bits.

33. (New) The method of claim 2, wherein the destination information includes exactly one bit.